

---

# **TECHNICKÁ UNIVERZITA V LIBERCI**

Fakulta mechatroniky, informatiky a mezioborových studií

Studijní program: N2612 – Elektrotechnika a informatika

Studijní obor: 1802T007 – Informační technologie

## **Systém pro evidenci školení zaměstnanců**

## **Employees training evidence system**

### **Diplomová práce**

Autor:

**Bc. Adam Truhlář**

Vedoucí práce:

Ing. Tomáš Martinec, Ph.D.

Konzultant:

Ing. Přemysl Svoboda

**V Liberci 17. 5. 2013**

# Prohlášení

Byl jsem seznámen s tím, že na mou diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., o právu autorském, zejména § 60 – školní dílo.

Beru na vědomí, že Technická univerzita v Liberci (TUL) nezasahuje do mých autorských práv užitím mé diplomové práce pro vnitřní potřebu TUL.

Užiji-li diplomovou práci nebo poskytnu-li licenci k jejímu využití, jsem si vědom povinnosti informovat o této skutečnosti TUL; v tomto případě má TUL právo ode mne požadovat úhradu nákladů, které vynaložila na vytvoření díla, až do jejich skutečné výše.

Diplomovou práci jsem vypracoval samostatně s použitím uvedené literatury a na základě konzultací s vedoucím diplomové práce a konzultantem.

V Liberci, 17.05.2013

Adam Truhlář

# Poděkování

Především bych rád poděkoval konzultantovi práce Ing. Přemyslu Svobodovi za poskytnuté informace a poskytnutý čas při vedení diplomové práce. Dále bych chtěl poděkovat TRW Lucas Varity s.r.o. a Radce Tomandlové za poskytnuté zdroje a možnost programovat aplikace.

# Abstrakt

Motivací pro vznik systému evidence školení zaměstnanců byla účast firmy TRW v projektu Operačního programu lidské zdroje a zaměstnanost, která vyžadovala tvorbu monitorovacích zpráv, které informují o počtech proškolených zaměstnanců, podpořených osob a jiných indikátorech. Pro evidenci školení zaměstnanců byl ve firmě TRW používán MS EXCEL, který byl nepřehledný, složitý na údržbu a neposkytoval požadované výstupy, proto vznikl požadavek na tvorbu nové aplikace evidence školení zaměstnanců.

Pro realizaci aplikace byla zvolena technologie .NET vzhledem k nasazenému operačnímu systému Windows. K tvorbě grafického rozhraní byla využita technologie Windows Forms. Data jsou ukládána v jedné ze dvou databází MS ACCESS nebo MS SQL Server Express. Komunikaci aplikace s databází zastřešuje technologie ADO.NET.

Vzniklá aplikace je nasazena ve firmě TRW, kde usnadňuje evidenci školení zaměstnanců a zároveň poskytuje potřebné výstupy pro monitorovací zprávy a exporty výstupů do dokumentů, které lze hromadně vytisknout.

## Klíčová slova

Programování;

Uživatelské grafické rozhraní;

Databáze;

Prezentace dat;

Evidence školení zaměstnanců.

# Abstract

The motivation for the development of the Employees training evidence system was TRW company participation in OP LZZ project, which required the creation of monitoring reports that provide information on the numbers of trained staff, supported people and other indicators. In TRW Company MS EXCEL was used as an employee evidence system, which was confusing, had complex maintenance and did not provide the desired outputs and so the new application was needed.

.NET technology was selected to create application due to the Windows operating system used in TRW Company. Windows Forms was selected as a GUI technology for application. Data is stored in one of two databases, MS Access or MS SQL Server Express. Application database communication is covered with ADO.NET technology.

The resulting application is deployed in the TRW company and provides evidence of employees training, necessary outputs for monitoring reports and exports to documents that can be printed in bulk.

## Keywords

Programming;

Graphical User Interface

Database;

Data presentation;

Employees training evidence.

# Obsah

Prohlášení.....	3
Poděkování.....	4
Abstrakt.....	5
Abstract .....	6
Seznam ilustrací .....	8
Seznam tabulek .....	9
Seznam použitých zkratk.....	10
Úvod.....	11
1 Rešerše současných systémů.....	13
2 Desktopové aplikace s GUI.....	16
2.1 Programování aplikací s GUI pro Windows .....	16
2.2 Windows Forms .....	19
3 Ukládání dat .....	22
3.1 Způsoby ukládání dat .....	22
3.2 Databázový systém .....	23
3.3 ADO.NET .....	26
4 Realizace .....	29
4.1 Monitorovací zprávy .....	29
4.2 Návrh aplikace .....	31
4.3 Návrh databáze.....	43
4.4 Komunikace aplikace s databází .....	45
4.5 Import zaměstnanců .....	47
4.6 Generování certifikátů a dohod.....	48
4.7 Hromadný tisk.....	51
4.8 Aktualizace aplikace .....	51
Závěr .....	54
Citace .....	56
Seznam příloh .....	58

# Seznam ilustrací

Obrázek 1: Architektura ADO.NET .....	28
Obrázek 2: DFD aplikace evidence školení zaměstnanců .....	32
Obrázek 3: Záložka školení .....	33
Obrázek 4: Katalog školení.....	33
Obrázek 5: Formulář zaměstnanců ve středisku .....	35
Obrázek 6: Záložka zaměstnanci .....	36
Obrázek 7: Záložka tvorba certifikátů .....	37
Obrázek 8: Formulář tvorby certifikátů .....	38
Obrázek 9: Záložka tvorba dohod.....	39
Obrázek 10: Záložka přehled realizovaných školení .....	40
Obrázek 11: Záložka podpořené osoby .....	42
Obrázek 12: Záložka zaměstnanci a školení.....	42
Obrázek 13: ERA diagram databáze školení zaměstnanců.....	44
Obrázek 14: Ukázka kódu pro vyplnění DataTable údaji z databáze .....	45
Obrázek 15: Formulář pro import zaměstnanců .....	47
Obrázek 16: Šablona pro tvorbu certifikátů.....	50
Obrázek 17: Ukázka kódu pro odeslání dokumentů na tiskárnu .....	51

# Seznam tabulek

Tabulka 1: Verze .NET Framework .....	17
Tabulka 2: Omezení MS ACCESS 2003 a SQL Server 2008 R2 Express.....	26
Tabulka 3: Rozdíly MS ACCESS 2003 a SQL Server 2008 R2 Express.....	26
Tabulka 4: Klíčové aktivity .....	31



# Seznam použitých zkratek

.NET - Vývojová platforma;

C# - C Sharp (Objektově orientovaný programovací jazyk);

CSV - Comma-separated value (hodnoty oddělené čárkou);

DFD - Data Flow Diagram;

ELM – Enterprise Learning Management;

ERA - Entity Relation with Attributes;

GNU - Projekt zaměřený na svobodný software;

GUI - Graphical User Interface (grafické rozhraní);

HR – Human Resources;

MS – Microsoft;

MSSQL - Microsoft SQL Server;

OP LZZ – Operační program Lidské zdroje a zaměstnanost

T-SQL – Transact SQL;

SQL - Structured Query Language;

VB.NET - Visual Basic .NET;

W3C – World Wide Web Consortium;

WPF - Windows Presentation Foundation;

XML - Extensible Markup Language.

# Úvod

Jakákoliv firma potřebuje do jisté míry uchovávat a zpracovávat informace o svých zaměstnancích. Čím více firma zaměstnává zaměstnanců, tím více potřebuje kvalitnější a robustnější nástroje pro správu těchto informací. Jedná-li se o firmu s několika zaměstnanci, postačí jako médium pro uchování informace obyčejný papír, který je zařazen v kartotéce. S rostoucím obsahem kartotéky se prodlužuje doba vyhledávání a zpracování informací a zároveň s ním roste i riziko výskytu chyb. Jako východisko z nastíněné situace firmy přecházejí na elektronické zpracování informací, které výše uvedené problémy řeší a navíc umožňuje jednoduché zálohování informací nebo napojení na jiné firemní systémy. Podle požadavků je navržen systém, který firmě vyhovuje - od ukládání informací v textovém souboru až po rozsáhlé databázové systémy. Pokud přestane stávající řešení vyhovovat je potřeba systém upravit nebo přijít se systémem novým.

Jedním takovým systémem byl i systém pro evidenci školení zaměstnanců ve firmě TRW. Firma TRW získala dotaci z projektu EU Školení je šance. V rámci tohoto projektu bylo základní potřebou získávání pololetních monitorovacích zpráv obsahujících různé indikátory, které je nutné dodržet. Firma uchovávala evidenci školení zaměstnanců v rozsáhlém dokumentu Microsoft Excel, který obsahoval spoustu propojených dat napříč jednotlivými listy. S novými zaměstnanci, školeními a požadavky na výpočet identifikátorů se dokument stal nepřehledný a zbytečně složitý pro práci.

Dokument v Microsoft Excelu nebylo možné jednoduše upravit a tak vznikl požadavek na nový systém, který by zjednodušil práci s evidencí školení zaměstnanců a poskytl potřebné monitorovací identifikátory. Výše uvedený systém je předmětem diplomové práce.

Při návrhu systému bylo nezbytné vyřešit několik otázek. Vytvořit webovou aplikaci nebo desktopovou aplikaci? Jakou použít technologii pro aplikace? Jak ukládat

informace? Jak zobrazit výstupy? Dle výše uvedených otázek lze rozdělit diplomovou práci na tři části. První částí je popis desktopových aplikací. Druhou část potom obsadí ukládání dat a poslední část popíše realizaci systému.

# 1 Rešerše současných systémů

Firmy s malým počtem zaměstnanců nejčastěji využívají pro správu zaměstnanců a školení Microsoft Excel. Zaměstnanci a školení jsou zde zapsáni ve formě tabulky. Nevýhodou MS EXCEL je, že 2 uživatelé nemohou pracovat na stejném listu zároveň, takže současně může evidenci školení zaměstnanců spravovat pouze jeden člověk. V údajích v MS EXCEL lze snadno udělat chybu, protože spoléhá na funkce, které jsou uvedeny v každé buňce zvlášť. Při kopírování a vkládání uživatel musí dát pozor, jaké vzorečky kopíruje a kam je kopíruje. Snadno tak dojde k chybě. Dalším problémem MS EXCEL, který může nastat při užívání více uživateli, je nepřehlednost. Uživatel, který dokument vytváří, se v něm snadno orientuje, to ale neznamená, že ostatní uživatelé, kteří jsou nuceni tento dokument využívat, se v dokumentu orientují také. To ve výsledku vede k dalším možným chybám v evidenci školení zaměstnanců.

Často ale firmy nevyužívají ani MS EXCEL ani jiný informační systém. Veškeré informace uchovávají v tištěné podobě.

Firmy s větším počtem zaměstnanců potom nasazují robustnější a komplexnější systémy, které neřeší pouze evidenci školení zaměstnanců, ale poskytují nejrůznější moduly od mezd, docházkové systémy až po systémy objednávací. Mezi využívané systémy patří např.:

- **Nugget** je vyvíjen českou firmou Nugget SW, s.r.o. sídlící v Praze. Systém Nugget má klient/server architekturu a poskytuje moduly pro mzdy, personalistiku a lidské zdroje. Evidenci školení zaměstnanců zajišťuje modul personalistika (Nugget SW, 2013)
- **PeopleSoft Enterprise Learning Management (ELM)** vyvíjen firmou Oracle. ELM je software určený pro plánování a evidenci školení včetně sledování platnosti periodických školení. ELM pro přístup zaměstnanců využívá webové aplikace. Výhody systému spočívají v přístupu pro zaměstnance i vedoucí a v detailních zprávách a přehledech o absolvovaných kurzech. Systém s sebou nese nevýhody v podobě složitosti vyplývající

z mnoha funkcí, nedostupnosti základní verze v českém jazyce, konzultace se správcem systému pouze telefonicky a pouze v angličtině. (Tomandlová, 2013)

- **FairSail** je komplexní personální systém vyvíjený firmou FairSail sídlící v Anglii. Systém je rozdělen do 4 částí: HR (Human Resources), HCM, Payflow a Recruit. Evidence školení umožňuje Modul Education v části HR. (Fairsail, 2013)

Uvedené firemní systémy poskytují rozsáhlé množství funkcí, modulů a konfigurací, aby mohly splnit co nejvíce požadavků firem. To je ale zároveň hlavním problémem těchto systémů. Uvedené systémy s rostoucí funkcionalitou přestávají být přehledné a stávají se složitější na pochopení a efektivní pro práci. Systémy jsou drahé a potřebují kvalitní podporu ze strany vývojáře, která je pro firmu často nedostatečná. Žádný z uvedených systémů neumožňuje tvorbu certifikátů ani neposkytuje všechny požadované výstupy pro monitorovací zprávy.

Proto vznikla aplikace evidence školení zaměstnanců, která je vytvořena na míru personálnímu oddělení ve firmě TRW. Neobsahuje žádné zbytečné funkce a je jednoduchá na pochopení a ovládání. Vývoj aplikace doprovázela úzká spolupráce s personálním oddělením, především s Ing. Radkou Tomandlovou, která je hlavní osobou podněcující vznik aplikace evidence školení zaměstnanců. Ing. Radka Tomandlová poskytovala výbornou zpětnou vazbu při testování nových funkcí a oprav chyb a přicházela s nápady pro zjednodušení práce s aplikací a dalšími rozšířeními funkcionality.

Vyvíjení interních aplikací ve firmě má své výhody a nevýhody. Hlavní výhodou je úzká spolupráce s vývojářem a definování požadavků na aplikaci. Při vývoji aplikace lze vyzkoušet různá řešení a nápady, z kterých je vybráno to nejlepší. Mezi další výhody se řadí snadné rozšíření o nové funkce a přímá podpora. Interně vyvíjené aplikace jsou levnější a zpravidla jednodušší na ovládání a pochopení protože nejsou zatíženy nepotřebnými funkcemi. Data interně vyvíjené aplikace je možno zpřístupnit i jiným aplikacím a tím snadno rozšířit funkcionalitu. Hotové aplikace také obsahují

méně chyb, protože jsou používány ostatními uživateli, kteří již případné chyby nahlásili a vývojáři pak nahlášené chyby opravili. Nevýhodou interně vyvíjených aplikací je časová náročnost uvedení aplikace do provozu, oproti hotovým aplikacím, které lze po nakonfigurování ihned používat.

## 2 Desktopové aplikace s GUI

Vysvětlení pojmu desktopové aplikace s GUI:

- Aplikační software (zkráceně aplikace) „umožňuje na počítači provádět nějakou užitečnou činnost (např. zpracování textů, výpočty, kreslení technických výkresů, zpracování obrazu, zvuku nebo videa).“ (Kolář, 2005, s. 2 )
- Pod pojmem desktopová aplikace se rozumí taková aplikace, která běží samostatně na počítači. Pojem desktopová aplikace kontrastuje s pojmem webová aplikace, která ke svému běhu vyžaduje webový prohlížeč.
- GUI (z angličtiny Graphical User Interface). Grafické uživatelské rozhraní umožňuje uživatelům komunikovat se stroji pomocí grafických interaktivních ovládacích prvků namísto psaní příkazů do příkazové řádky. Dobře navržené uživatelské rozhraní zjednodušuje uživateli interakci s aplikací.

Desktopové aplikace s GUI jsou tedy samostatně běžící aplikace, které uživateli poskytují grafické rozhraní pro interakci s aplikací.

### 2.1 Programování aplikací s GUI pro Windows

V současné době existuje několik programovacích jazyků, které lze využít pro programování aplikací s GUI pro Windows. V aplikaci evidence školení zaměstnanců bylo využito technologie Microsoft .NET a programovacího jazyka VB.NET, protože ve firmě TRW se v personálním oddělení využívá operační systém Windows a ke spuštění .NET aplikace je potřeba .NET Framework, který je součástí Windows. Není tedy potřeba instalovat další software pro spuštění aplikace.

Microsoft .NET je vyvíjen firmou Microsoft především pro operační systém Windows, který je nasazen ve firmě TRW. Základní komponentou .NET je .NET Framework, prostředí potřebné pro běh aplikací a nabízející jak spouštěcí rozhraní, tak potřebné knihovny. Historie .NET frameworku sahá až do roku 2002, kdy byla vydána první generace nesoucí označení 1.0. (Thai et al., 2003) Chronologicky seřazené verze .NET frameworku viz tabulka 1.

**Tabulka 1: Verze .NET framework**

Verze	Datum vydání	Distribuce s OS
1.0	2002	Bez distribuce s OS
1.1	2003	Windows Server 2003
2.0	2005	Windows Server 2003 R2
3.0	2006	Windows Vista a Windows Server 2008
3.5	2007	Windows 7 a Windows Server 2008 R2
4.0	2010	Bez distribuce s OS
4.5	2012	Windows 8 a Windows Server 2012

Aplikace evidence školení zaměstnanců je postavena na .NET frameworku 3.5, protože je součástí operačního systému Windows 7. V současné verzi .NET framework umožňuje vytvářet aplikace všeho druhu od webových aplikací až po grafické aplikace využívající DirectX. Existuje GNU obdoba .NET, která se nazývá DotGNU (GNU je projekt zaměřený na svobodný software). DotGNU se stará o přenositelnost celé platformy, tedy umožňuje spouštět všechny .NET aplikace na unixových platformách (Linux, Mac OS X, Solaris). Programátor píšící .NET aplikace není omezen volbou jednoho programovacího jazyka. Bez ohledu na to v čem byla aplikace napsána se vždy přeloží do mezijazyka Common Intermediate Language. K programování na platformě .NET se nejčastěji používá IDE Microsoft Visual Studio, které je v současnosti k dispozici ve dvou variantách. První placenou variantu lze nalézt pod názvem Microsoft Visual Studio a druhá neplacená varianta nese název Microsoft Visual Studio Express. Visual Studio Express 2012 se dále dělí podle nasazení na následující produkty:



- **Visual Studio Express 2012 for Web** poskytuje možnost vývoje webových aplikací.
- **Visual Studio Express 2012 for Windows 8** umožňuje vývoj aplikací pro Windows 8 s uživatelským rozhraním Metro.
- **Visual Studio Express 2012 for Windows Desktop** umožňuje tvorbu klasických desktopových aplikací. Windows XP není podporován.
- **Visual Studio Team Foundation Server Express 2012** mezi hlavní rysy patří kontrola nad zdrojovým kódem, sledování jednotlivých pracovních úkolů a správa při tvorbě aplikace pro skupinu až pěti vývojářů.
- **Visual Studio Express 2012 for Windows Phone** zpřístupňuje tvorbu aplikací pro mobilní operační systém Windows Phone. Vzniklé mobilní aplikace lze otestovat pomocí přiloženého emulátoru. (Microsoft, 2013a)

Pro tvorbu desktopových aplikací s GUI na platformě .NET lze použít jednu ze dvou možností, které jsou součástí .NET frameworku.

- **Windows Forms** zastupují vývojově starší možnost. Mezi hlavní rysy patří jednoduchost a množství funkčně obsáhlých komponent pro zobrazování obsahu. Pokud by nabídka komponent programátorovi nestačila, tak si může vytvořit komponentu vlastní, čehož využívají některé firmy a poskytují nespočet dalších placených a neplacených komponent.
- **WPF** (Windows Presentation Foundation) je součástí .NET frameworku od verze 3.5. Pro vykreslování komponent je využito technologie DirectX, která umožňuje zobrazení složitých komponent a dokáže některé grafické požadavky převést na grafickou kartu a tím snížit zatížení procesoru. WPF podporuje vektorovou grafiku důležitou pro vykreslení zmenšených nebo zvětšených komponent bez ztráty kvality obrazu. Tvorba GUI je umožněno jazykem XAML, který je založen na XML. Pomocí XAML jsou popsány a propojeny jednotlivé komponenty. (Microsoft, 2013b) Kromě Visual Studia lze pro návrh GUI použít např. Microsoft Expression Blend, který generuje XAML kód, který lze následně použít v aplikaci.

## 2.2 Windows Forms

Vzhledem k faktu, že pro tvorbu GUI aplikace evidence školení zaměstnanců bylo využito technologie Windows Forms, tak bude tato technologie podrobněji popsána. Každá UI komponenta Windows Forms je odvozena od třídy *Control*. Nejvýše v hierarchii komponent stojí komponenta *Form* (formulář). Protože každá komponenta je odvozena od třídy *Control*, sdílejí jednotlivé komponenty vlastnosti společné třídě *Control*. Tato dědičnost zajišťuje jistou minimální funkcionalitu napříč všemi UI komponentami. Mezi tyto sdílené vlastnosti patří velikost a pozice komponenty, reakce na vstupy a vzhled. Třída *Control* také definuje vztah mezi jednotlivými komponentami umístěnými ve formuláři. Podporována je vazba předek-potomek, takže komponenta může obsahovat několik potomků komponent. Tato vazba ovlivňuje všechny komponenty pro dané operace jako přesunování a změna velikosti. (Griffiths, 2003)

Návrh grafického rozhraní je velmi jednoduchý. Programátor si nejdříve vytvoří nový formulář a nastaví mu požadované vlastnosti. Vytvoření formuláře ve Visual Studiu automaticky vygeneruje příslušný kód, který formulář definuje. V dalším kroku poté na formulář přesune požadované komponenty a opět nastaví vlastnosti. S každou změnou vlastností a novou komponentou ve formuláři se vygeneruje příslušný kód. Tímto je tvorba grafického rozhraní hotova a zbývá dopsat kód, který bude obsluhovat jednotlivé komponenty. Následující seznam obsahuje nejpoužívanější komponenty Windows Forms.

- **Button** slouží ke spuštění různých akcí např. uložení informací na formuláři.
- **CheckBox** může být volitelně dvoustavové nebo třístavové zaškrtačkové pole.
- **ComboBox** pomůže k výběru jedné konkrétní položky ze seznamu, který je přiřazen komponentě *ComboBox*. Seznam položek *ComboBoxu* je v základním stavu skrytý, pro zobrazení více položek je potřeba kliknout na symbol šipky směřující dolů. Je-li seznam obsáhlý, lze urychlit nalezení požadované položky vypsáním počátečních písmen hledaného výrazu.
- **DateTimePicker** se použije v případě kdy je zapotřebí vybrat konkrétní datum. Uživateli je při výběru data zobrazen malý měsíční kalendář s názvy

dnů v týdnu. Pokud není definováno jinak tak se jako výchozí datum v kalendáři nastaví aktuální datum operačního systému.

- **Label** umožňuje zobrazit textový řetězec ve formuláři.
- **LinkLabel** podobně jako *Label* zobrazí text, ale s tím rozdílem, že text bude zvýrazněn a po kliknutí na text provede akci.
- **ListBox** stejně jako *ComboBox* má přiřazen seznam textových položek s tím rozdílem, že zobrazuje najednou více položek a umožňuje výběr jedné nebo více položek.
- **ListView** je rozšířením komponenty *ListBox*. Namísto textu lze zobrazit obrázky nebo i obrázky s textem. Položky *ListView* lze seskupovat do skupin a přiřazovat další potomky.
- **RadioButton** neboli radiové tlačítko. Nachází-li se v kontejneru formuláře více těchto komponent, tak právě jedna z nich může být vybrána. Při výběru jiného *RadioButton* se automaticky předchozí odznačí.
- **RichTextBox** zpřístupňuje uživateli psaní a zobrazení formátovaného textu.
- **TextBox** je odlehčenou variantou *RichTextBox*, tedy postrádá podporu formátovaného textu a podporuje pouze nezformátovaný text.
- **Tooltip** zobrazuje nápovědu po přejetí myši nad komponentami ve formě vyskakovací bubliny.
- **TreeView** implementuje stromovou strukturu položek. Položky v *TreeView* lze označovat, zaškrtnávat, skrývat a zobrazovat pomocí předků.
- **WebBrowser** jak název napovídá, implementuje jednoduchý webový prohlížeč.
- **Chart** najde uplatnění tam, kde je zapotřebí zobrazit grafy
- **DataGridView** je jednou z nejdůležitějších komponent. Připojená data zobrazuje jako tabulku. Sloupce *DataGridView* mohou obsahovat obrázky, text, tlačítka, *ComboBox*, *CheckBox*, a další komponenty. Data obsažená v *DataGridView* mohou být filtrována pomocí příslušných hlaviček sloupců, mazána, upravována nebo přidávána

Mezi komponentami lze nalézt tzv. kontejnerové komponenty, které usnadňují rozmístění komponent a jejich vizuální oddělení.

- **FlowLayoutPanel** seskupuje vnořené komponenty do horizontálního nebo vertikálního směru. Každá přidaná komponenta je zařazena za poslední vloženou komponentu.
- **GroupBox** zobrazí rám okolo skupiny vnořených komponent s možností přidání titulků rámu. Vhodný pro grafické oddělení skupiny komponent.
- **Panel** je komponenta podobná *GroupBox* s tím rozdílem, že neobsahuje rám a nelze přiřadit titulek rámu.
- **SplitContainer** rozdělí zobrazenou plochu formuláře pomocí oddělovače. Rozdělení plochy může být horizontální nebo vertikální. S oddělovačem může uživatel posouvat dle potřeby nebo nemůže, pokud je nastavena vlastnost *IsSplitterFixed* na hodnotu *True*.
- **TabControl** se používá pro záložkovou funkcionalitu. Skupiny komponent jsou přiřazeny jednotlivým záložkám s názvy, mezi kterými se lze přepínat kliknutím na název záložky. Všechny záložky sdílejí stejnou velikost. Přepnutím záložky se stará záložka s komponentami skryje a je překreslena komponentami z nové záložky.
- **TableLayoutPanel** umožní rozvržení komponent do tabulky s definovanými sloupci a řádky. Každý sloupec nebo řádek může mít definovanou fixní velikost nebo poměrnou velikost v procentech vzhledem k tabulce. Případně sloupec nebo řádek může mít nastavenou vlastnost *autosize*, kdy je velikost automaticky určena tak, aby umožnila zobrazit celou vnořenou komponentu.

Poslední uvedenou komponentou je **OpenFileDialog**, kterou lze uživateli usnadnit zápis cesty k souboru nebo souborům. Pomocí metody *ShowDialog()* se zobrazí systémové okno průzkumníku Windows, ve kterém si uživatel označí potřebné soubory. Pomocí vlastnosti *Filter* lze nastavit filtr pro zobrazení souborů. Filtrování souborů probíhá na základě přípon souborů.

## 3 Ukládání dat

Kapitola ukládání dat je zaměřena na možnosti ukládání dat na počítačích a manipulaci s nimi. K tomu jsou využívány databáze a technologie umožňující přístup k databázím. Databáze a přístup, který byl využit při návrhu aplikace, je popsán v podkapitolách.

### 3.1 Způsoby ukládání dat

Důležitou součástí aplikace evidence školení zaměstnanců je způsob jak se ukládají informace. Nejpoužívanějším způsobem pro jednoduché ukládání informací je ukládání do souboru. Existuje nepřeberné množství souborových formátů a každý má specifické využití od ukládání prostého textu až po ukládání videa. V rámci evidence školení zaměstnanců je potřeba ukládat strukturovaná data o zaměstnancích a školení.

Jedním ze způsobů jak ukládat strukturovaná data do souborů je vytvoření si vlastního souborového formátu, který bude splňovat programátorovy požadavky. Pokud se programátor nechce zatěžovat tvorbou vlastního souborového formátu, potom může využít některý z již existujících formátů:

- **CSV** (z angličtiny comma-separated values) ukládá informace do souboru v textové formě tak, že jednotlivé hodnoty jsou odděleny středníkem. Řádky hodnot jsou odděleny oddělovačem řádku. Formát se často používá pro ukládání tabulek, kdy na prvním řádku jsou uvedeny názvy sloupců oddělené středníkem a ve zbylých řádcích jsou uloženy hodnoty. (Shafranovich, 2005)
- **XML** (z angličtiny Extensible Markup Language) definuje formátování dokumentu tak, aby bylo dobře čitelné pro člověka i pro stroj. Cílem XML je jednoduchost, obecnost a použitelnost na internetu. XML dokument se skládá z elementů a atributů. Element je tvořen počátečním a koncovým tagem. Dokument XML musí obsahovat pouze jeden kořenový element, další elementy musí být vnořené do kořenového elementu. Programátor si

pomocí pravidel XML vytvoří tzv. značkovací jazyk, který lze následně využít pro uložení informací. (W3C, 2012)

### 3.2 Databázový systém

Databázový systém, také nazývaný database management systém (DBMS) je obecně software pro manipulování s databázemi. Jedná se v podstatě o počítačový systém pro uchovávání záznamů a umožňuje uživatelům přidávat, mazat, měnit a získávat a aktualizovat tyto informace dle potřeby. Databázový systém umožňuje souběžné používání databáze více uživateli. DBMS je také souhrn programů, které umožňují uživateli databáze vytvářet a spravovat. (Singh, 2011)

Současný trh nabízí hned několik databázových systémů:

- MySQL - nejrozšířenější databázový systém díky otevřenosti jeho softwaru a licencování GPL i komerčnímu licencování;
- MSSQL - alternativní řešení od společnosti Microsoft;
- PostgreSQL - další otevřený systém vyvíjený primárně pro Linux;
- Oracle - moderní multiplatformní databázový systém od společnosti Oracle Corporation.

MSSQL (Microsoft SQL server) je jedním ze dvou databázových systému, který může aplikace evidence školení zaměstnanců využít pro ukládání informací. Existuje několik různých verzí, které pokrývají různé požadavky uživatelů. Historie MS SQL Serveru sahá až do roku 1989, kdy vznikla verze SQL Server 1.0. Nejaktuálnější verze nese označení SQL Server 2012. Pro komunikaci s aplikací byla zvolena verze MS SQL 2008 R2 Express SP2, která je jedinou bezplatnou verzí. Zvolení verze Express s sebou nese jistá omezení:

- omezení velikosti jedné databáze na 10 GB;
- využití pouze jednoho procesoru;
- využití maximálně 1 GB operační paměti RAM;
- bez pokročilých možností a nástrojů vyšších verzí. (Microsoft, 2013c)

Pro zjednodušení práce s MSSQL databází vydal Microsoft nástroj SQL Server Management Studio (ssms), který byl poprvé spuštěn s verzí SQL serveru 2005. Uvedený nástroj poskytuje nástroje pro nastavení, správu všech komponent SQL Serveru. Dále obsahuje automatizovaný nástroj pro tvorbu databázových diagramů, editor skriptů a další grafické nástroje pro práci s objekty a funkcemi.

Microsoft Access zastupuje druhou možnost pro ukládání informací z aplikace školení zaměstnanců. Microsoft Access je dalším databázovým systémem od společnosti Microsoft obsažený v placeném kancelářském balíku aplikací Microsoft Office, který kombinuje GUI s databází. MS ACCESS uchovává data ve vlastním formátu založeném na Access Jet Database Engine, který umožňuje importovat nebo propojit informace z jiných aplikací nebo databází. Vlastní data jsou uložena v souborech s příponou „.mdb“ nebo „.accdb“. MS ACCESS podporuje objektově orientovaný jazyk Visual Basic for Application pro tvorbu aplikací a poskytuje nástroje pro tvorbu formulářů a jiných výstupů. Název Visual Basic for Application může být matoucí, protože je podobný názvu jazyka Visual Basic .NET, ale jedná se o jiný programovací jazyk, který není s Visual Basic .NET kompatibilní. Program napsaný ve Visual Basic for Application potom může být přiřazen tlačítku, makru nebo klávesové zkratce.

Důvodem, proč aplikace evidence školení zaměstnanců podporuje dva různé databázové systémy, je rozdílné zaměření obou systémů. MS ACCESS byl vyvinut s cílem, aby poskytl jednoduchý nástroj pro tvorbu relačních databází a zároveň umožnil tvorbu jednoduchých formulářů a výstupů pro manipulaci s daty uloženými v databázi. K uvedenému cíli dopomáhá intuitivní GUI a množství dialogových pomocníků (průvodců) pro tvorbu a nastavení databáze. Obsah celé databáze s formuláři, makry a dalšími objekty je uložen v jediném souboru, který lze snadno přenášet. Pro otevření

souboru potřebuje uživatel mít na svém PC nainstalován Microsoft Access, který je nainstalovaný na počítačích ve firmě TRW.

SQL Server Express na rozdíl od souborového modelu MS ACCESS používá klient/server model a neposkytuje žádné nástroje pro tvorbu formulářů a výstupů pro manipulaci s daty. Distribuce SQL Server Express databáze nespočívá pouze v přesunutí jednoho souboru. Pro distribuci SQL Server Express databáze je potřeba PC s nainstalovaným a nakonfigurovaným SQL Server Express nebo s vyšší verzí. Potom lze využít zálohování databáze, které uloží obsah databáze do souboru. Následně se na cílovém počítači použije nástroj pro obnovení databáze ze souboru. Způsobů, jak distribuovat SQL Server Express databázi, existuje více. Lze využít generování skriptů v Microsoft SQL Server Management Studio, které vygeneruje skript pro vytvoření zvolené databáze. Vygenerovaný skript se potom spustí na cílovém PC a vytvoří požadovanou databázi.

SQL Server Express se dále odlišuje robustností. Neomezuje počet připojených uživatelů ani počet vytvořených databází. Při výměně dat mezi SQL Server Express a aplikací po síti jsou kladeny nižší požadavky na propustnost sítě a nedochází k porušení databáze při výpadku, na který je náchylná MS ACCESS databáze. Pokud je soubor databáze MS ACCESS zpřístupněn více uživatelům, tj. soubor je uložen na sdíleném síťovém úložišti, potom s uvedeným souborem může pracovat více uživatelů najednou. Microsoft stanovil maximální počet uživatelů pro práci s MS ACCESS databází na hodnotu 255. V praxi se ale ukazuje, že i menší počet uživatelů způsobuje problémy v databázi. Čím větší počet uživatelů pracuje s jednou databází MS ACCESS, tím jsou kladeny vyšší požadavky na propustnost a stabilitu sítě. Odezva na požadavky uživatelů se prodlužuje natolik, až se databáze stává nepoužitelnou. Možnosti nasazení MS ACCESS a SQL Server 2008 R2 Express vyplývají z omezení (viz tabulka 2).



**Tabulka 1: Omezení MS ACCESS 2003 a SQL Server 2008 R2 Express**

Omezení	SQL Server 2008 R2 Express	MS ACCESS 2003
Počet současně připojených uživatelů	32767	255
Velikost databáze	10GB	2GB
Počet využitých procesorů	1	1
Využití paměti	1GB	2GB

**Tabulka 2: Rozdíly MS ACCESS 2003 a SQL Server 2008 R2 Express**

Rozdíly	SQL Server 2008 R2 Express	MS ACCESS 2003
Model	Klient/server	Souborový
Cena	Zdarma	Zdarma
Nástroje pro tvorbu formulářů	Ne	Ano
Dotazovací jazyk	T-SQL	Kompatibilní s SQL 92

MS ACCESS 2003 a SQL Server 2008 R2 Express používají odlišné dotazovací jazyky. Dotaz napsaný pro MS ACCESS nemusí být validní pro Microsoft SQL Server a naopak. Aplikaci, která komunikuje s MS ACCESS a Microsoft SQL Server, nestačí komunikovat pouze jedním společným dotazovacím jazykem, ale musí se dotazovat dvěma odlišnými dotazovacími jazyky.

### 3.3 ADO.NET

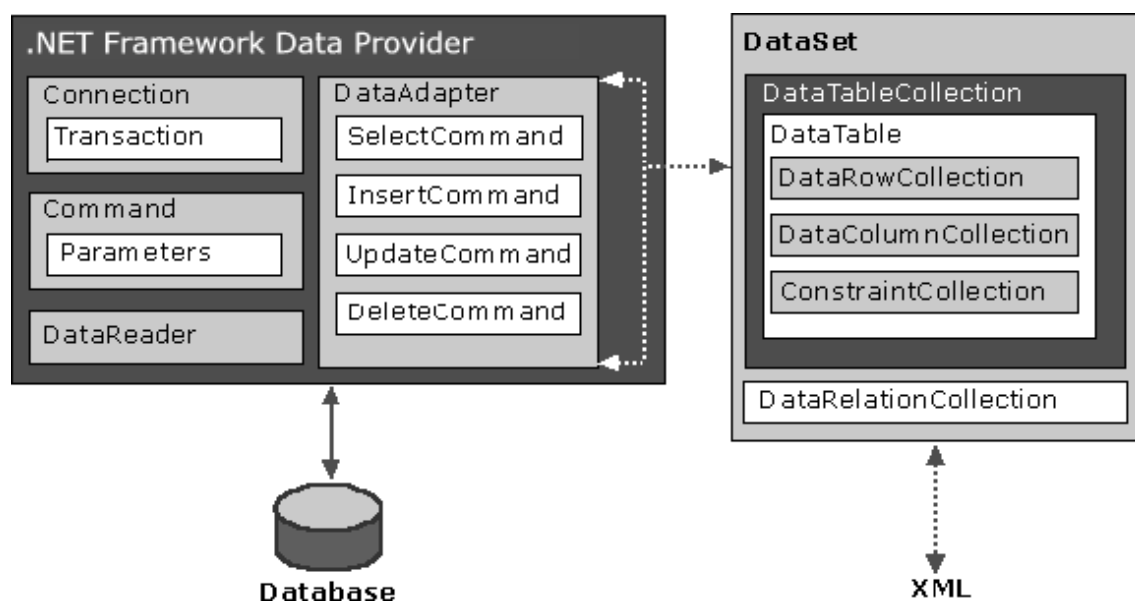
Pro usnadnění programování aplikací na platformě .NET využívajících databáze připravil Microsoft technologii ADO.NET. ADO.NET obsahuje řadu tříd, které poskytují přístup k datům a datovým službám založených na *DataSetech* a XML.

ADO.NET je součástí .NET framework. ADO.NET odděluje přístup k datům od manipulace s daty do komponent. ADO.NET dále obsahuje .NET Framework data providers pro připojení k databázi, vykonávání příkazů a získávání výsledků. Výsledky jsou přímo zpracovávány nebo umístěny do ADO.NET *DataSet* objektu pro předložení uživateli. *DataSet* objekt může být také využíván nezávisle na .NET Framework data provider ke správě lokálních dat aplikace nebo k načtení dat z XML. ADO.NET poskytuje nejpřímější cestu k přístupu dat v .NET frameworku. Pro využití vyšší abstrakce lze využít ADO.NET Entity Framework. (Microsoft, 2013d)

.NET Framework Data Provider se skládá z několika částí.

- **Connection** objekt poskytuje připojení ke zdroji dat.
- **Command** objekt umožňuje přístup k databázovým dotazům pro návrat dat, manipulaci s daty, spouštění procedur a odesílání nebo přijímání informací o parametrech.
- **DataReader** poskytuje výkonný tok dat ze zdroje dat.
- **DataAdapter** je mostem mezi zdrojem dat a *DataSet*. Využívá *command* objektu k vykonávání SQL dotazů k naplnění *DataSetu* daty a k zaslání změn v *DataSetu* zpět do zdroje dat.

*DataSet* byl navržen tak, aby se mohl využít nezávisle s různými zdroji dat. Ve výsledku může být *DataSet* použit s různými databázemi, XML daty nebo ke správě dat v aplikaci. *DataSet* je složen z kolekce jedné nebo více *DataTable* objektů, který obsahuje kolekci *DataRow* objektů, *DataColumn* objektů a také primárních klíčů, cizích klíčů. V *DataSetu* jsou také uloženy informace mezi vazbami jednotlivých *DataTable* objektů. Ilustrace ADO.NET architektury viz obrázek 1. (Microsoft, 2013e)



**Obrázek 1: Architektura ADO.NET**

Zdroj: Microsoft, 2013e

## 4 Realizace

Kapitola realizace popisuje jednotlivé funkcionality aplikace evidence školení zaměstnanců a výchozí požadavky monitorovacích zpráv, na jejichž základě byla aplikace navržena.

### 4.1 Monitorovací zprávy

Aplikace evidence školení zaměstnanců vznikla pro usnadnění získávání výstupů pro monitorovací zprávy. Monitorovací zprávy jsou pro firmu TRW důležité, protože firma TRW je příjemcem finanční podpory z Operačního programu lidské zdroje a zaměstnanost. Firma TRW se při žádosti o finanční podporu zavázala, že bude mít 1090 úspěšných absolventů kurzů a 700 podpořených osob. Monitorovací indikátory (ukazatele) poskytují zpětnou vazbu o tom, zda poskytnutá finanční podpora splnila svůj účel, tedy že projekty dosáhly cílů, které firma TRW uvedla v žádosti o finanční podporu. Při monitorování realizace projektu musí firma TRW naplňování těchto ukazatelů průběžně sledovat a vykazovat v monitorovacích zprávách projektu. Vykazované hodnoty u těchto závazných indikátorů musí být prokazatelné a ověřitelné případnou kontrolou.

Následující seznam popisuje potřebné indikátory výstupů:

- **Počet podpořených osob** je celkový počet osob, které v rámci projektu získaly jakoukoliv formu podpory bez ohledu na počet poskytnutých podpor. Podpora má nejčastěji formu vzdělávacího nebo rekvalifikačního kurzu, školení apod. Každá podpořená osoba je započítávána pouze jednou, bez ohledu na to, kolik podpor obdržela.
- **Počet podpořených osob – muži.**
- **Počet podpořených osob – ženy.**

- **Počet podpořených osob – mladí lidé 15-24 let.**
- **Počet podpořených osob – starší pracovníci 55-64 let.** (Leistner, 2010)

Dále jsou počty podpořených osob odlišovány podle vzdělání:

- **Počet podpořených osob – základní ISCED 1 a 2** uvádí počet osob, které mají základní nebo nižší střední vzdělání.
- **Počet podpořených osob – střední ISCED 3** uvádí počet osob se středním vzděláním.
- **Počet podpořených osob – nástavbové studium ISCED 4** určuje počet osob s kategorií dosaženého vzdělání na nástavbovém studiu.
- **Počet podpořených osob – vysokoškolské ISCED 5 a 6** určuje počet osob s vysokoškolským vzděláním. (Leistner, 2010)

Poslední skupinu indikátorů tvoří následující indikátory:

- **Počet úspěšných absolventů kurzů celkem** definuje počet osob, které získaly kvalifikaci/doklad o kvalifikaci. Jedná se o osoby, které ukončily kurz způsobem stanoveným poskytovatelem kurzu (např. získaly certifikát). Konkrétní účastník je započítán tolikrát, kolik kurzů úspěšně dokončil předepsaným způsobem.
- **Počet úspěšných absolventů kurzů – muži.**
- **Počet úspěšných absolventů kurzů – ženy.** (Leistner, 2010)

Výše uvedené indikátory je nutné sledovat k určitému datu, protože monitorovací zprávy jsou sbírány pololetně. *„Dosažené hodnoty indikátorů se v monitorovacích zprávách uvádějí kumulativně, tedy vždy souhrnně za období od počátku projektu do konce příslušného monitorovacího období.“* (Leistner, 2010, s. 5)

Každé školení v rámci projektu Operačního programu lidské zdroje a zaměstnanost spadá pod určenou klíčovou aktivitu, která má své unikátní číslo. Firma TRW definovala záměry v rámci projektu pro všechny klíčové aktivity. Všechny klíčové aktivity a jejich čísla jsou uvedeny (viz tabulka 4).

**Tabulka 3: Klíčové aktivity**

Číslo aktivity	Název klíčové aktivity
01	Řízení lidských zdrojů
02	Vzdělávání zaměstnanců nad 50 let
03	Odborné a profesní vzdělávání
04	Vzdělávání v oblasti kvality
05	Operation Excellence & Lean aktivity
06	Vzdělávání zaměstnanců na mateřské dovolené
07	Odborná jazyková příprava
08	Vzdělávání interních lektorů
09	Tvorba e-learningových vzdělávacích programů
10	Řízení projektu

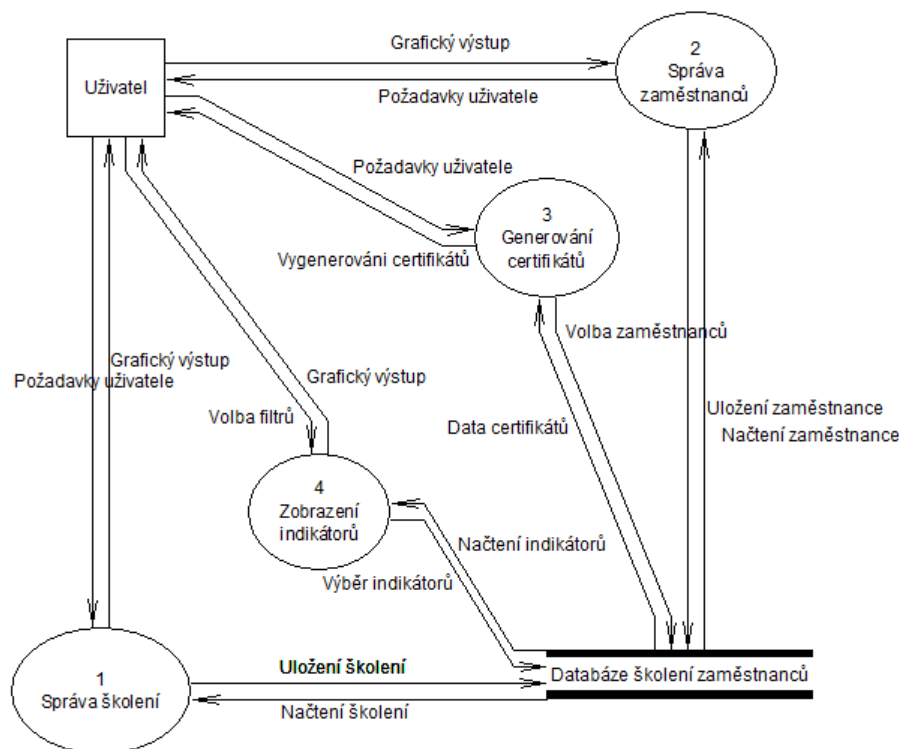
Zdroj: Leistner, 2010

## 4.2 Návrh aplikace

Použití aplikace zachycuje Data Flow Diagram (viz obrázek 2).

Aplikaci využívají tři lidé současně, kteří aktualizují zaměstnance ve firmě, přidávají nové školení do katalogu školení, plánují školení pro zaměstnance, zařazují zaměstnance do školení, zapisují úspěšné absolventy školení a vytvářejí certifikáty. Uvedený datový tok je znázorněn na výše zobrazeném Data Flow Diagramu.

Základní okno aplikace je koncipováno ze sedmi záložek (školení, zaměstnanci, tvorba certifikátů, tvorba dohod, přehled realizovaných školení, podpořené osoby, zaměstnanci a školení) v komponentě *TabControl*, které pokrývají všechnu potřebnou funkcionalitu aplikace. Jednotlivým záložkám jsou věnovány následující podkapitoly. V základním okně jsou rovněž přístupné informace o aktuální i předchozí verzi aplikace společně s provedenými změnami. Pro podporu lze využít komponenty *LinkButton* s textem podpora v pravém horním rohu aplikace, které otevře novou zprávu ve zvoleném poštovním klientovi a vyplní informace o adresátovi a předmět zprávy.



**Obrázek 2: DFD aplikace evidence školení zaměstnanců**

#### 4.2.1 Záložka školení

Záložka školení (viz obrázek 3) poskytuje přístup ke katalogu školení, plánování školení a zapisování zaměstnanců do školení a jejich úspěšného absolvování školení.

Záložka školení je logicky rozdělena na čtyři části obalené komponentami *GroupBox* nesoucí názvy (možnosti, všechna školení, detaily školení a účastníci školení). V oblasti „Možnosti“ má uživatel přístup ke katalogu školení, kde jsou školení zařazena do klíčových aktivit a mají přiřazené evidenční číslo. Ukázka katalogu školení viz obrázek 4. V horní části katalogu školení má uživatel přístup k vytvoření nového školení. K novému školení musí uživatel zadat evidenční číslo, které musí být unikátní v rámci celého katalogu, název školení a nakonec je nutné nové školení zařadit do číselníku klíčových aktivit. Po vyplnění všech informací lze uložit nové školení do katalogu školení stisknutím tlačítka přidej. Přidané školení je zařazeno do komponenty

*DataGridView* obsahující seznam školení, kde lze vytvořená školení nadále upravovat nebo mazat.

**Detaily školení**

Aktivita: 1 - Řízení lidských zdrojů

Evidenční číslo: Back to Basis

Certifikát: ☒

Produkt: ☐

Název produktu:

Datumy: 9/11/2012 - 9/14/2012 [Přidat/upravit/zobrazit datumy](#)

Testovací hodnocení:

Poznámka:

Ukončeno: ☐

[Uložit změny](#) [Odstranit školení](#)

**Účastníci školení**

Přidat zaměstnance do školení: [pomocí střediska](#)

Začněte psát příjmení zaměstnance:

Úspěšný absolvent: [zaškrtnout všem](#) [odebrat všem](#)

	Jméno	Úspěšný absolvent	Osobní číslo	PS ID	Profese	
▶	Truhlář Ad...	<input checked="" type="checkbox"/>	1111	1111	analýza sys...	<input checked="" type="checkbox"/>
	Starý Josef	<input checked="" type="checkbox"/>	3333	3333	Zdravotní s...	<input checked="" type="checkbox"/>
	Stará Josef...	<input checked="" type="checkbox"/>	4444	4444	CF - OIP5	<input checked="" type="checkbox"/>
	Nová Petra	<input checked="" type="checkbox"/>	2222	2222	logistika - di...	<input checked="" type="checkbox"/>

**Obrázek 3: Záložka školení**

**Katalog školení**

Nové školení

ev. č. 1 - Řízení lidských zdrojů  [Přidej](#)

	ev_c	Název	Aktivita	
▶	1	Komplexní nákupčí od A a...	1 - Řízení lidských zdrojů	<input checked="" type="checkbox"/>
	2	SQL Obchodní vyjednávání	1 - Řízení lidských zdrojů	<input checked="" type="checkbox"/>
	3	Komplexní nákupčí od A a...	6 - Vzdělávání zaměstnanc...	<input checked="" type="checkbox"/>
	4	Komplexní nákupčí od A a...	3 - Odborné profesní vzděl...	<input checked="" type="checkbox"/>
	6	Lektorské minimum	8 - Vzdělávání interních lek...	<input checked="" type="checkbox"/>
	7	Pracovní instruktáž dle met...	8 - Vzdělávání interních lek...	<input checked="" type="checkbox"/>
	8	Kvalita ve výrobě	4 - Vzdělávání v oblasti kv...	<input checked="" type="checkbox"/>
	9	Systém 11 barev pro výrob...	4 - Vzdělávání v oblasti kv...	<input checked="" type="checkbox"/>
	10	SPC pro buřiče a měňče ...	4 - Vzdělávání v oblasti kv...	<input checked="" type="checkbox"/>
	11	Obsluha profiloměru	4 - Vzdělávání v oblasti kv...	<input checked="" type="checkbox"/>

**Obrázek 4: Katalog školení**



Vybráním termínu a školení z uvedeného katalogu se pomocí tlačítka „Nové školení“ naplňuje školení.

Školení může mít jeden nebo více termínů. Naplánované školení se zobrazí v části „Všechna školení“, kde lze naplánovaná školení seřadit podle data, evidenčního čísla nebo názvu školení. V části „Všechna školení“ je u názvu školení vždy uvedeno v závorce evidenční číslo a datum začátku školení. Pořadí názvu školení, evidenčního čísla a data začátku u naplánovaných školení se mění podle toho, jakým způsobem jsou školení seřazena. V části „Všechna školení“ jsou pro větší čitelnost barevně odlišena ukončená a neukončená školení. Nedokončené školení má červený text a dokončené školení má text zelený.

Kliknutím na naplánované školení se v části „Detail školení“ zobrazí podrobnosti o vybraném školení, které je možno dále upravovat. Uživatel u jednotlivých školení určuje výstup certifikátu ze školení nebo jiný produkt, upravuje termíny školení, připsuje poznámky a zaškrťává, jestli zvolené školení bylo ukončeno. Při provedení změn v naplánovaném školení je nutné upravené informace uložit kliknutím na tlačítko „Uložit změny“. Naplánované školení lze smazat z databáze kliknutím na tlačítko „Odstranit školení“.

Zařazení účastníků do školení umožňuje část „Účastníci školení“. Uživatel má na výběr dva způsoby jak zařadit zaměstnance do školení. Prvním způsobem lze zařadit zaměstnance do školení využitím formuláře „Zaměstnanci ve středisku“ (viz obrázek 5). Formulář Zaměstnanci ve středisku obsahuje komponentu ComboBox, která slouží pro výběr střediska ze seznamu všech středisek. Po zvolení nebo vepsání čísla střediska se v přílehlé komponentě DataGridView zobrazí zaměstnanci, kteří pracují pod určeným střediskem. Do školení lze zařadit všechny zaměstnance z určeného školení kliknutím na tlačítko „Přidat všechny“ nebo lze vybrat ze seznamu zaměstnanců řádky zaměstnanců, které lze přidat kliknutím na tlačítko „Přidat označené“. Přidání zaměstnanci jsou v pozadí zařazeni do účastníků školení v záložce školení. Aby nedocházelo k nechtěnému mazání účastníků školení ze záložky školení, zatímco je otevřen formulář zaměstnanci ve středisku, tak je formulář zaměstnanci ve středisku

vyvolán pomocí metody „ShowDialog()“, která znemožní jakoukoliv uživatelskou interakci s ostatními formuláři, dokud není vyvolaný formulář ukončen. Ukončit formulář zaměstnanci ve středisku lze kliknutím na křížek v pravém horním rohu formuláře. Druhý způsob přidávání zaměstnanců do školení využívá našeptávače. Uživatel začne psát příjmení zaměstnance a našeptávač zobrazuje shodný výběr zaměstnanců s osobním číslem, z kterého je zvolen správný zaměstnanec.

	Jméno	Os.číslo	PS ID
▶	Truhlář Adam	1111	1111
	Starý Josef	3333	3333
	Stará Josefína	4444	4444
	a ab	132	123

**Obrázek 5: Formulář zaměstnanců ve středisku**

U přiřazených účastníků školení lze nastavit úspěšného absolventa. Pro zjednodušení označení nebo odznačení úspěšných absolventů je poskytnuta možnost hromadného označení a pomocí podtrženého textu „zaškrtnout všem“ a „odebrat všem“.

#### 4.2.2 Záložka zaměstnanci

Pro správu zaměstnanců je připravena záložka zaměstnanci (obrázek 6). K dispozici jsou tlačítka pro vytvoření nového zaměstnance a pro import zaměstnanců ze souboru csv. V části „Seznam zaměstnanců“ je využito komponenty *GridView* pro editaci a mazání zaměstnanců. Tabulka zaměstnanců má fixní sloupec se jménem zaměstnance a tlačítkem se symbolem křížku pro smazání zaměstnance. V tabulce zaměstnanců jsou uvedeny všechny údaje o zaměstnancích, které lze kliknutím na buňku tabulky upravit.

Pro rychlejší vyhledání zaměstnanců je připraven filtr, který při psaní příjmení zaměstnance schovává zaměstnance, kteří neodpovídají hledanému příjmení. Pro psaní příjmení zaměstnance byla naprogramována vlastní komponenta pojmenovaná „TextBoxWatermark“, která dědí od standartní komponenty *TextBox* a přidává novou funkcionalitu. Komponenta „TextBoxWatermark“ zobrazuje programátorem definovaný text: „Začněte psát příjmení zaměstnance“, který je zobrazen, dokud uživatel nezačne psát příjmení zaměstnance.

	Jméno	Doú	Datum narození	Bydliště	Vznik p.s.	Os. číslo	ps_id
▶	a ab	<input checked="" type="checkbox"/>	3/5/2013	ab	3/5/2013	132	123
	Jan Les0	<input type="checkbox"/>	1/16/1983	Nám Dr E.X 3/4, ...	1/3/2012	651932	0
	Nová Petra	<input checked="" type="checkbox"/>	9/11/1975	Jablonec	9/11/2012	2222	2222
	Nový Jaroslav	<input checked="" type="checkbox"/>	9/11/1974	Jablonec	9/11/2012	6666	6666
	Stará Josefína	<input type="checkbox"/>	9/11/1970	Jablonec	9/11/2012	4444	4444
	Starý Josef	<input type="checkbox"/>	9/11/1960	Jablonec	9/11/2012	3333	3333
	Truhlář Adam	<input type="checkbox"/>	9/11/1987	Jablonec	9/11/2012	1111	1111
	Voda Martin Ing.	<input type="checkbox"/>	8/1/1989	1.Máje 125, 539 ...	1/3/2012	632716	0

**Obrázek 6: Záložka zaměstnanci**

#### 4.2.3 Záložka tvorba certifikátů

Obdobně jako v záložce školení je v levé části formuláře seznam školení v komponentě *TreeView* s rozdílem, že jsou zobrazena pouze ta školení, která mají jako definovaný výstup certifikát. Školení lze seřadit podle data, evidenčního čísla nebo názvu. Formát názvu školení a barevné zvýraznění je shodné se seznamem školení na záložce školení. Na rozdíl od *TreeView* v záložce školení má *TreeView* v záložce tvorba certifikátů

změněnou vlastnost „CheckBoxes“, která přidává funkcionalitu pro označení školení pro tvorbu certifikátů. Při kliknutí na školení v seznamu se školení vybere a zobrazí svůj název v části „Vybrané školení“. U vybraného školení se v části „Účastníci školení“ zobrazí tabulka účastníků školení. Kliknutím na buňky v prvním sloupci v tabulce účastníků školení se označují zaměstnanci pro tvorbu certifikátů. Ukázka záložky tvorba certifikátů viz obrázek 7.

	Jméno	Osobní číslo	PS ID	F
<input checked="" type="checkbox"/>	Truhlář Adam	1111	1111	a.
<input type="checkbox"/>	Starý Josef	3333	3333	Z.
<input type="checkbox"/>	Stará Josefína	4444	4444	C.
<input type="checkbox"/>	Nová Petra	2222	2222	I...

**Obrázek 7: Záložka tvorba certifikátů**

Uživatel má při tvorbě certifikátů na výběr tři možnosti jak definovat, které certifikáty chce vytisknout:

- **Vytvořit všechny certifikáty** – vygeneruje certifikáty pro všechny zaměstnance ve všech školeních, které mají definovaný výstup certifikát.
- **Vytvořit všechny certifikáty pro vybrané školení** – vytvoří certifikáty pro všechny zaměstnance v označených školeních v komponentě *TreeView*.
- **Vytvořit certifikáty pro označené zaměstnance** – vytvoří certifikáty pro označené účastníky vybraného školení.

Výše uvedené možnosti jsou zpřístupněny pomocí tlačítek, které mají shodný název s uvedenými možnostmi. Po kliknutí na vybranou možnost pro tvorbu certifikátů se zobrazí následující formulář (obrázek 8).

Tvorba certifikátů

Šablony pro generování certifikátů

N:\School\IDP - aplikace\Skoleni\Skoleni\bin\x86\Debug\Skoleni.xml

Změnit

Kam uložit vytvořené certifikáty

N:\School\IDP - aplikace\Skoleni\Skoleni\bin\x86\Debug

Změnit

☒ Uložit certifikáty do podsložek podle názvu školení.

Certifikátů k vytvoření: 4

:Vytvořeno 0

Vytisknout Vytvořit certifikáty

**Obrázek 8: Formulář tvorby certifikátů**

Na formuláři tvorby certifikátů si uživatel zvolí šablonu pro certifikát ve formátu XML a umístění, kam chce vytvořené certifikáty uložit. Na výběr je i možnost ukládání certifikátů do podsložek podle názvu školení. Formulář dále obsahuje informaci o počtu certifikátů k vytvoření a počtu vytvořených certifikátů. Stisknutím tlačítka „Vytvořit certifikáty“ se určené certifikáty vytvoří do specifikovaného umístění. Pokud uživatel nechce vytvářet certifikáty, může je pomocí tlačítka „Vytisknout“ poslat na tiskárnu. Stisknutí tlačítka „Vytisknout“ zobrazí standardní dialogové okno pro tisk dokumentů na platformě Windows, kde lze zvolit tiskárnu pro tisk certifikátů. O průběhu tvorby certifikátů je uživatel informován komponentou „ProgressBar“, která spustí svou animaci při tvorbě certifikátů a která se zastaví po dokončení tvorby certifikátů. Při tvorbě certifikátů se průběžně mění hodnota u počtu vytvořených certifikátů. Po úspěšném dokončení tvorby certifikátů je uživateli zobrazena informace: „Hotovo, všechny soubory byly vytvořeny“. Nastane-li chyba při tvorbě certifikátů, tak je o ní uživatel rovněž informován.

#### 4.2.4 Záložka tvorba dohod

Záložka tvorba dohod (viz obrázek 9) není hlavní požadavkem aplikace, ale protože databáze obsahuje všechny důležité informace a zároveň umožňuje tvorbu certifikátů, bylo navrženo implementovat tuto dodatečnou funkcionalitu pro tvorbu dohod o účasti zaměstnance v projektu. Záložka je rozdělena do dvou logických částí: výběr zaměstnanců a vybraní zaměstnanci. Uživatel si při tvorbě dohod o účasti v projektu vybírá, pro které zaměstnance chce vytvořit nebo vytisknout dohodu. Vybraní zaměstnanci jsou zobrazeni v tabulce v části „Vybraní zaměstnanci“. Vybírat zaměstnance pro dohodu lze několika způsoby:

- **Nedávno přidání zaměstnanci** – zobrazí formulář pro výběr zaměstnanců ze seznamu. V seznamu je posledních 20 zaměstnanců, kteří byli naposled přidáni do databáze.
- **Podle střediska** – pomocí formuláře umožní uživateli vybrat všechny nebo část zaměstnanců, kteří spadají pod určené středisko.
- **Pomocí příjmení zaměstnance** – uživatel při výběru zaměstnance postupuje shodně jako při zařazování zaměstnanců do školení.
- **Pomocí osobního čísla zaměstnance** - uživatel začne psát osobní číslo zaměstnance a našeptávač zobrazuje shodný výběr zaměstnanců s osobním číslem, z kterého je zvolen správný zaměstnanec.

	Jméno	Os. č.	Bydliště	Narození
▶	Truhlář Adam	1111	Jablonec	9/11/1987
	Stará Josefína	4444	Jablonec	9/11/1970
	Voda Martin Ing.	632716	1.Máje 125, 539 ...	8/1/1989
	Nová Petra	2222	Jablonec	9/11/1975

Obrázek 9: Záložka tvorba dohod

#### 4.2.5 Záložka přehled realizovaných školení

Záložka přehled realizovaných školení (viz obrázek 10) zobrazuje přehled o školeních podle zvolených filtrů:

- **Datum od začátku školení;**
- **Datum do konce školení;**
- **Účastník;**
- **Aktivita;**
- **Konkrétní školení;**
- **Katalog školení;**
- **Včetně neukončených školení.**

Popis  
Zobrazí údaje o školeních podle zvolených filtrů. [Zobrazit/schovat filtry](#)

Filtry

☒ Datum od: 4/13/2010 do: 3/13/2013

☐ Účastník

☐ Středisko

☐ Aktivita

☐ Konkrétní školení

☐ Katalog školení

☒ Zobrazit včetně neukončených

Data

	Název školení	Datum od	Datum do	Certifikát	Produkt	Název produktu	Počet účastníků	M
▶	(27) - Back to Basis	9/11/2012	9/14/2012	<input checked="" type="checkbox"/>	<input type="checkbox"/>		4	2
	(22) - Efektivní komunikace...	9/11/2012	9/13/2012	<input type="checkbox"/>	<input type="checkbox"/>		3	2
	(19) - Efektivní vedení týmu...	9/20/2012	3/12/2013	<input checked="" type="checkbox"/>	<input type="checkbox"/>		3	1
	(2) - Obchodní vyjednávání	3/5/2013	3/5/2013	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	ACCESS	0	0
	(33) - ACCESS Interkulturáln...	3/5/2013	3/9/2013	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	asd	3	2
	<b>CELKEM</b>			<input type="checkbox"/>	<input type="checkbox"/>		<b>13</b>	<b>7</b>

Obrázek 10: Záložka přehled realizovaných školení

Pro zahrnutí filtru do přehledu realizovaných školení je nutné konkrétní filtr povolit zaškrtnutím komponenty *CheckBox* s názvem filtru. Uživatel není omezen výběrem jednoho filtru, ale může si povolit filtry dle svých potřeb. Pro usnadnění zadávání data ve filtru „Datum od začátku školení“ a „Datum do konce školení“ je využito komponenty *DateTimePicker*. Po zvolení filtrů lze načíst data o realizovaných školeních kliknutím na tlačítko s textem „Načíst data“. Načtená data jsou zobrazena v tabulce ve spodní části s názvem „Data“. Tabulka přehledu realizovaných školení poskytuje informace o názvu školení, poznámce, datum od, datum do, certifikátu, produktu, počtu účastníků, počtu žen a mužů, počtu účastníků podle kategorie vzdělání a počtu účastníků podle věkové kategorie. Na posledním řádku tabulky se nachází souhrnný řádek, v kterém jsou uvedeny součty pro sloupce obsahující počty účastníků.

#### 4.2.6 Záložka podpořené osoby

Záložka podpořené osoby (viz obrázek 11) informuje uživatele o počtu podpořených zaměstnanců ke zvolenému datu a zároveň uvádí počty jednotlivých zaměstnanců rozdělených podle věku, dosaženého vzdělání a pohlaví. Uvedené informace o podpořených osobách se nacházejí v části „Podpořené osoby“, kde lze zároveň zvolit, ke kterému datu se mají spočítat podpořené osoby. Podpořené osoby se spočítají po kliknutí na tlačítko „Načti údaje“ a zobrazí se v tabulce v části „Výpočet podpořených osob“. Tabulka obsahuje sloupce: jméno, osobní číslo, středisko, profese, typ vzdělání, účastí celkem, úspěšná absolvování a podporující školení.



Školení | Zaměstnanci | Tvorba certifikátů | Tvorba dohod | Přehled realizovaných školení | Podpořené osoby | Zaměst

Podpořené osoby

k datu: 9/14/2012

**Podpořených osob celkem: 2**

**Podle vzdělání:**

s věkem < 24:	0	Mužů:	1	Základní ISCED 1 a 2	0
s věkem > 24 a < 50:	2	Žen:	1	Střední ISCED 3	0
s věkem > 50:	0			Nástavbové ISCED 4	0
s věkem > 55 a < 64:	0			Vysokoškolské ISCED 5 a 6	2

Výčet podpořených osob

	Jméno	Osobní číslo	Středisko	Profese	Typ vzdělání	Účastí celkem
▶	Truhlář Adam	1111	1000	analýza systému j...	ISCED 5 Bc.	1
	Stará Josefína	4444	1000	CF - OIP5	ISCED 6 Vysoko...	1
	<b>CELKEM</b>					<b>2</b>

Obrázek 11: Záložka podpořené osoby

#### 4.2.7 Záložka zaměstnanci a školení

Záložka zaměstnanci a školení (viz obrázek 12) zobrazuje v tabulce všechny zaměstnance ve školení podle zvolené klíčové aktivity a data. Dále poskytuje informace o počtu úspěšných a neúspěšných zaměstnanců a o počtu zaměstnanců v ukončených a neukončených školeních.

Tvorba certifikátů | Tvorba dohod | Přehled realizovaných školení | Podpořené osoby | Zaměstnanci a školení

Výběr aktivity

!Všechna školení

K datu: 4/13/2013

☒ Zobrazit včetně neukončených školení

Neúspěšných absolventů: 2  
 Úspěšných absolventů: 13  
 Zaměstnanců v neukončených školeních: 10  
 Zaměstnanců v ukončených školeních: 5

Zaměstnanci ve školení

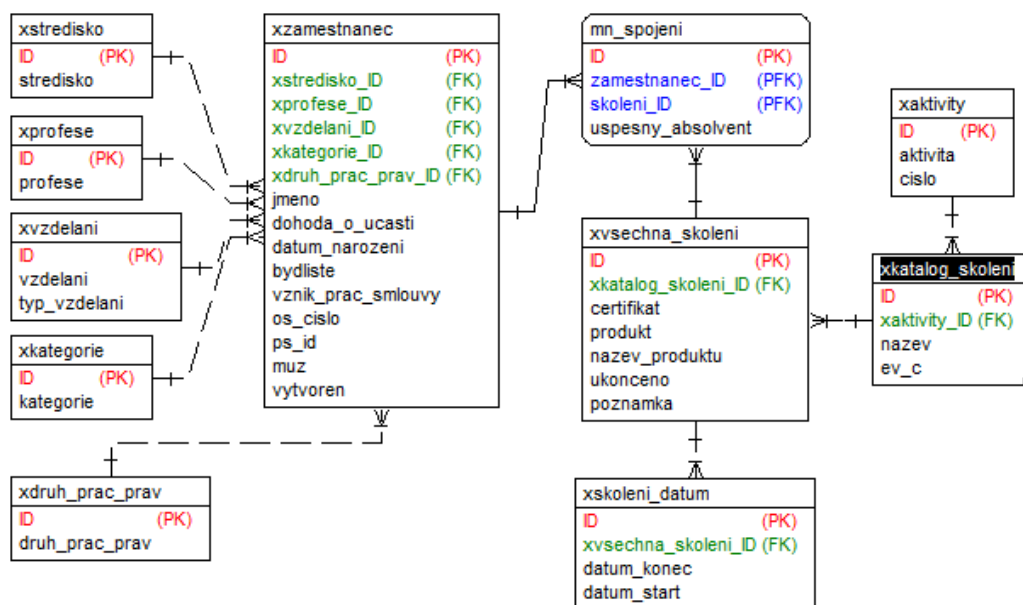
	Aktivita	Úspěšný absolvent	Č. aktivity	Školení	Jméno	Osobní č.
▶	Řízení lidských z...	<input checked="" type="checkbox"/>	1	ACCESS Interkult...	Stará Josefína	4444
	Řízení lidských z...	<input checked="" type="checkbox"/>	1	ACCESS Interkult...	Starý Josef	3333
	Řízení lidských z...	<input checked="" type="checkbox"/>	1	ACCESS Interkult...	Truhlář Adam	1111
	Řízení lidských z...	<input checked="" type="checkbox"/>	1	Back to Basis	Nová Petra	2222
	Řízení lidských z...	<input checked="" type="checkbox"/>	1	Back to Basis	Stará Josefína	4444

Obrázek 12: Záložka zaměstnanci a školení

Uživatel má možnost volby zobrazení včetně neukončených školení. Údaje jsou načteny do tabulky v části „Zaměstnanci ve školení“. Jeden řádek v tabulce podává informaci o jednom zaměstnanci v jednom školení. Tabulka obsahuje sloupce: aktivita, úspěšný absolvent, číslo aktivity, název školení, jméno zaměstnance, osobní číslo, středisko, účastí v aktivitě a školení ukončeno. Řádky mají zelený text, pokud obsahují ukončené školení a naopak červený text pro neukončené školení. Úspěšný absolvent je v tabulce zvýrazněn žlutým podbarvením buňky ve sloupci úspěšný absolvent. Pokud absolvent není úspěšný, je buňka ve sloupci úspěšný absolvent podbarvena modře.

### 4.3 Návrh databáze

Návrh databáze viz obrázek 13. Aplikace pracuje se zaměstnanci a školeními, které tvoří základní entity *xzamestnanec* a *xkatalog\_skoleni*. Každé školení má svůj název a evidenční číslo a zároveň má přiřazenou klíčovou kategorii. Klíčové kategorie jsou realizovány entitou *xaktivita*. Klíčová aktivita je složena z názvu a čísla. Entita *xkatalog\_skoleni* je použita na naplánování školení. Naplánované školení zastupuje entita *xvsechna\_skoleni*, která obsahuje atribut certifikát definující, že výstupem ze školení je certifikát. Atribut produkt v entitě *xvsechna\_skoleni* informuje o produktu školení. Pokud existuje produkt školení, pak je název produktu uchován v atributu *nazev\_produkту*. Pro psaní poznámek k naplánovanému školení slouží atribut *poznámka*. Ukončené školení je definováno atributem *ukonceno*.



**Obrázek 13: ERA diagram databáze školení zaměstnanců**

Školení nemusí probíhat pouze v jednom termínu, ale může být rozděleno na vyšší počet nespojitých časových intervalů. Proto nestačí mít pouze jednu dvojici atributů datum\_od a datum\_do v entitě *xvsechna\_skoleni*. Aby mohlo mít naplánované školení více termínů tak byla vytvořena entita *xskoleni\_datum*, která se odkazuje cizím klíčem *xvsechna\_skoleni\_ID* na naplánované školení a obsahuje informace o začátku a konci termínu. Entita *xskoleni\_datum* umožňuje nadefinování více termínů pro jedno naplánované školení.

V naplánovaném školení může být více účastníků a zároveň jeden účastník může být ve více naplánovaných školeních. Uvedená relace je relací M:N mezi entitami *xzamestnanec* a *xvsechna\_skoleni*. Tato relace je realizována pomocí entity *mn\_spojeni*, v které je obsažena informace o úspěšném absolventovi atributem *uspesny\_absolvent*.

Každý zaměstnanec má jméno, bydliště, datum narození, dohodu o účasti, datum vzniku pracovní smlouvy, osobní číslo, *ps\_id*, pohlaví a datum zapsání do databáze. Na entitu *xzamestnanec* jsou navázány entity *xstredisko*, *xprofese*, *xvzdelani*, *xkategorie*, *xdruh\_prac\_prav*, které omezují nežádoucí redundanci dat pro informace o středisku, ve

kterém zaměstnanec pracuje, jakou profesi vykonává, jaké má vzdělání, do jaké kategorie spadá a jaký má druh pracovněprávního vztahu.

Entita *xvzdelani*, kromě názvu vzdělání, informuje o typu vzdělání, tj. určuje kategorie ISCED 1 až ISCED 6.

## 4.4 Komunikace aplikace s databází

Pro komunikaci aplikace školení zaměstnanců je využito již pospané technologie ADO.NET umožňující objektovou práci s relačními daty. Pro grafický výstup a manipulaci s daty využívá aplikace nejčastěji komponentu *DataGridView*, která společně s komponentami *ComboBox* a *ListBox* sdílí atribut *DataSource*. Atributu *DataSource* lze přiřadit dynamickou strukturu *DataTable* reprezentující získané informace z databáze v podobě tabulky. Vytvoření objektu *DataTable* a jeho vyplnění údaji z databáze ilustruje následující ukázka kódu viz obrázek 14.

```
Dim conn_str As String = _
    "Data Source=ADAM-NTB\MSSQL2008R2;Initial Catalog=db_skoleni;User ID=xxx;Password=xxx"
Dim DT As New DataTable
Using connection As New SqlConnection(conn_str)
    Using sql_da As New SqlDataAdapter("SELECT * FROM xzamestnanec", connection)
        sql_da.Fill(DT)
    End Using
End Using
```

**Obrázek 14: Ukázka kódu pro vyplnění DataTable údaji z databáze**

Pro získání informací z databáze je potřeba znát SQL dotaz a connection string obsahující údaje o umístění databáze a poskytovateli. V ukázce kódu jsou data získávána z databáze MSSQL 2008 R2 Express nacházející se na počítači ADAM-NTB pojmenované MSSQL20008R2. Protože v MSSQL se může nacházet více databází, je definována jako výchozí databáze databáze db\_skoleni parametrem Initial Catalog. V connection string se nachází přihlašovací údaje pro přístup k databázi.

V ukázce kódu (viz obrázek 14) je SQL dotaz reprezentován textem `SELECT * FROM xzamestnanec`. Uvedený SQL dotaz získá všechny údaje o zaměstnancích z tabulky *xzamestnanec*. Symbol „\*“ nahrazuje názvy všech sloupců, které jsou v tabulce *xzamestnanec* definovány.

SQL dotaz a connection string se použijí v konstruktoru *SqlDataAdapter*, který načte údaje z databáze a zapíše údaje do objektu *DataTable*. Takto získaný objekt *DataTable* lze přiřadit atributu *DataSource* komponenty *DataGridView*. Pokud má *DataGridView* nastavenou vlastnost *AutoGenerateColumns* na hodnotu *true* a nastavený *DataSource*, pak jsou všechny údaje z *DataSource* zobrazeny uživateli. V komponentě *DataGridView* je umožněno uživateli manipulovat s údaji nebo údaje seřadit podle hodnot ve vybraném sloupci. Změna dat v komponentě *DataGridView* je reflektována v přiřazeném objektu *DataTable*. Pro nahrání změn v objektu *DataTable* zpět do databáze je nutné použít metodu *Update* objektu *SqlDataAdapter*. Metoda *Update* požaduje dodefinování SQL dotazů v *SqlDataAdapter* pro vložení, aktualizaci a mazání záznamů pomocí atributů *InsertCommand*, *UpdateCommand* a *DeleteCommand*.

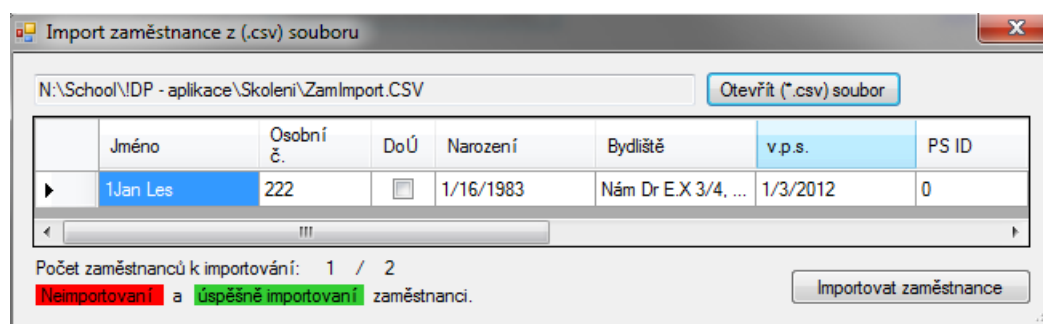
Komunikace s databází MS ACCESS probíhá na stejném principu. Liší se connection string, kde je uvedena cesta k mdf souboru databáze MS ACCESS a místo objektů *SqlConnection* a *SqlDataAdapter* jsou využívány objekty *OleDbConnection* a *OleDbDataAdapter*, kterými lze taktéž získat objekt *DataTable*. Pro komunikaci s databází MS ACCESS bylo potřeba přepsat některé dotazy použité pro MSSQL databázi, protože je využíváno odlišné syntaxe jazyka SQL.

V programování přístupu ke dvěma různým databázím bylo využito abstraktní třídy definující společné rozhraní pro komunikaci s databázemi. Rozhraní obsahuje všechny metody a funkce, které jsou využívány pro komunikaci. Ke komunikaci s vybranou databází pak bylo potřeba doprogramovat třídu implementující společné rozhraní, tj. upravit SQL dotazy v závislosti na použité databázi.

## 4.5 Import zaměstnanců

Import zaměstnanců usnadňuje přidávání nových zaměstnanců do databáze. Ve firmě TRW je využíváno více informačních systému, které již potřebné údaje o zaměstnancích obsahují a poskytují export zaměstnanců do souboru csv. Díky této funkcionalitě není nutné každého nového zaměstnance se všemi důležitými informacemi zdlouhavě vytvářet.

K importu zaměstnanců byl naprogramován formulář s jednoduchým ovládáním. Formulář (obrázek 15) je přístupný ze záložky zaměstnanci po kliknutí na tlačítko s nápisem „Importovat zaměstnance z (.csv)“.



**Obrázek 15: Formulář pro import zaměstnanců**

Formulář obsahuje dvě tlačítka. Tlačítkem „Otevřít (\*.csv) soubor“ je umožněno uživateli vybrat pomocí komponenty *OpenFileDialog* soubor, který má být použit pro import zaměstnanců. CSV soubor pro import zaměstnanců musí obsahovat předem definované názvy sloupců na prvním řádku a údaje o zaměstnancích v ostatních řádcích. Hodnoty v CSV souboru musí být odděleny středníkem. Po zvolení souboru je obsah souboru načten do objektu *DataTable* a zobrazen v komponentě *DataGridView*. Zobrazení jsou pouze ti zaměstnanci, kteří nejsou uloženi v databázi. Celkový počet zaměstnanců v souboru CSV a počet nových zaměstnanců je uveden za textem „Počet zaměstnanců k importování“. Rozpoznávání zaměstnanců pro přidání do databáze není řízeno jménem, ale osobním číslem. Načtené údaje v *DataGridView* lze dále upravit

před importem do databáze. Připravené údaje o zaměstnancích se importují kliknutím na tlačítko „Importovat zaměstnance“. O úspěšně importovaném zaměstnanci je uživatel informován zeleným podbarvením příslušného řádku. Pokud import zaměstnance selže, je řádek zaměstnance podbarven červeně.

## 4.6 Generování certifikátů a dohod

GUI pro tvorbu certifikátů a dohod je popsáno v kapitole 4.2.3 Záložka tvorba certifikátů a 4.2.4 Záložka tvorba dohod. Dynamicky generovat dokumenty lze několika způsoby v závislosti na tom, v jakém formátu má být výstupní dokument. Nejjednodušší pro tvorbu dokumentů je formát obyčejného textového dokumentu, kde lze zapsat textové řetězce přímo do souboru. Pro certifikát bohužel formát obyčejného textového dokumentu nedostačuje, a proto bylo zvoleno, že certifikáty budou uloženy ve formě dokumentu Microsoft Word, který lze snadno vytvořit a patřičně naformátovat.

K přístupu MS Word dokumentu na platformě .NET lze využít knihovny Microsoft Word Object Library, který poskytuje rozhraní pro manipulaci MS Word dokumentu. Bohužel tato knihovna je značně neefektivní pro hromadnou tvorbu dokumentů, protože pro každý nově vytvořený dokument se na pozadí Windows spustí proces „WINWORD.EXE“ reprezentující instanci programu MS WORD. Rychlé spouštění většího množství procesů vyžaduje spoustu systémových prostředků. Tvorba stovek certifikátů může trvat až několik minut. Dalším problémem je nutnost nainstalování MS Word, bez kterého knihovna Microsoft Word Object Library nefunguje.

Z výše uvedených důvodů bylo přistoupeno k jinému přístupu tvorby certifikátů. Tento přístup využívá faktu, že MS Word dokument lze uložit ve formátu XML. Práce s XML soubory na platformě .NET je několikanásobně rychlejší než použití Microsoft Word Object Library pro MS Word dokument. Pro manipulaci s XML soubory lze využít jmenného prostoru „System.Xml“, který je součástí .Net Framework.

Základními objekty pro manipulaci s XML souborem jsou:

- **XmlDocument** - mapuje XML soubor a obsahuje metody a atributy pro jednoduchou manipulaci s XML souborem. Mimo jiné obsahuje atribut *ChildNodes* typu *XmlNodeList*.
- **XmlNodeList** - je objekt, který obsahuje seznam elementů v dané úrovni a má referenci na předka typu *XmlNode* a potomky typu *XmlNodeList*.
- **XmlNode** - představuje jeden vybraný element XML souboru se všemi jeho atributy.

Pro vyhledávání elementů a atributů pomocí různých kritérií v *XmlDocument* lze použít dotazovací jazyk XPATH. Další možností jazyka XPATH je výpočet hodnot např. textových řetězců nebo čísel. Jazyk XPATH byl definován World Wide Web Consorciem (W3C).

Tvorba certifikátu probíhá tak, že si uživatel vytvoří šablonu jako dokument MS Word. V místech dokumentu, kde je potřeba vyplnit údaje z databáze, vloží uživatel typ pole MERGEFIELD a vyplní název pole podle informace, kterou chce v místě pole získat z databáze. Aplikace evidence školení zaměstnanců při tvorbě certifikátů otevře vytvořenou šablonu MS WORD dokumentu uloženou jako XML a pro všechny vybrané zaměstnance ve vybraných školeních vytvoří nový dokument MS Word. V novém dokumentu aplikace vyhledá všechny názvy polí MERGEFIELD pomocí jazyka XPATH a nahradí je příslušnou hodnotou z databáze. Hodnoty z databáze pro generování certifikátu jsou předány v konstruktoru pro formulář tvorby certifikátů objektem *DataTable*. Jako názvy polí MERGEFIELD lze tedy využít jakýkoliv název sloupce, který je součástí předávaného objektu *DataTable*. Pro každý řádek objektu *DataTable* je vytvořen právě jeden certifikát. Uvedený postup umožňuje tvorbu stovek certifikátů v rámci sekund.





TRW Automotive Czech s.r.o.  
 Braking Operations 421  
 Technical Centre 969  
 Jablonec nad Nisou

# CERTIFIKÁT

«jmeno»

«ve\_dnech»

absolvoval/a vzdělávací aktivitu financovanou  
z prostředků Evropského sociálního fondu prostřednictvím  
Operačního programu Lidské zdroje a zaměstnanost  
a státního rozpočtu ČR

«nazev»

Projekt  
 CZ.1.04/1.1.02/35.014/22  
 INOVACE NEZABRZDÍME

.....  
podpis koordinátora



evropský  
sociální  
fond v ČR



EVROPSKÁ UNIE



MPSV



OPERAČNÍ PROGRAM  
LIDSKÉ ZDROJE  
A ZAMĚSTNANOST

PODPORUJEME  
VAŠI BUDOUCNOST  
[www.esfcr.cz](http://www.esfcr.cz)

**Obrázek 16: Šablona pro tvorbu certifikátů**

Příklad šablony pro tvorbu certifikátů viz obrázek 16. Na obrázku šablony budou nahrazeny pole MERGEFIELD s názvy „<jmeno>“, „<ve\_dnech>“ a „<nazev>“ údaji z databáze obsahující jméno zaměstnance, termín školení a název školení z katalogu školení. Vytvořené certifikáty mají název souboru ve formátu „([evidenční číslo školení])\_[jméno zaměstnance]\_[datum začátku školení]“. Pokud již takový soubor se shodným názvem ve složce existuje, pak je na začátek názvu souboru přidáno číslo takové, které společně se zbytkem názvu školení tvoří unikátní název souboru ve složce.

Tvorba dohod využívá shodného postupu jako při tvorbě certifikátů. Pouze je využito jiné šablony s jinými nadefinovanými názvy MERGEFIELD a jiného odpovídajícího objektu DataTable. Vytvořené soubory dohod nesou název ve formátu „[jméno zaměstnance]\_[osobní číslo]“.

## 4.7 Hromadný tisk

Hromadný tisk probíhá ve dvou fázích. Soubory dokumentů se nejdříve vytvoří jako při tvorbě certifikátů a dohod a po dokončení se uživateli zobrazí nabídka *PrintDialog*, která obsahuje nabídku dostupných tiskáren. Po zvolení tiskárny se každý vytvořený soubor pošle na vybranou tiskárnu.

Volba tiskárny v komponentě *PrintDialog* vrátí požadovaný název tiskárny, který je použit při odeslání souboru na tiskárnu. Odeslání souboru na tiskárnu je realizováno pomocí parametrického spuštění procesu, který je reprezentován objektem *ProcessStartInfo*, který je součástí .NET Framework. Odeslání souborů zobrazuje ukázka kódu (viz obrázek 17).

```
If PrintDialog1.ShowDialog() = Windows.Forms.DialogResult.OK Then
    For Each f As String In files
        Dim info As New System.Diagnostics.ProcessStartInfo(f)
        info.Arguments = Chr(34) & PrintDialog1.PrinterSettings.PrinterName.ToString() & Chr(34)
        info.CreateNoWindow = True
        info.WindowStyle = System.Diagnostics.ProcessWindowStyle.Hidden
        info.UseShellExecute = True
        info.Verb = "PrintTo"
        Try
            System.Diagnostics.Process.Start(info)
        Catch ex As Exception
            MessageBox.Show(ex.ToString)
        Exit For
    End Try
Next
End If
```

**Obrázek 17: Ukázka kódu pro odeslání dokumentů na tiskárnu**

## 4.8 Aktualizace aplikace

Aplikace evidence školení zaměstnanců si před každým spuštěním kontroluje ze serveru, zda není přístupná novější verze aplikace. Pokud novější verze aplikace existuje, pak je ze serveru stažena a automaticky aktualizována. Uvedená funkcionality

je výhodná pro rychlé řešení vzniklých problémů a zároveň zajišťuje, že na všech počítačích, které provozují aplikaci je spuštěna aktuální verze.

K aktualizaci aplikace bylo využito technologie ClickOnce, která je součástí .NET Framework od verze 2.0. ClickOnce umožňuje nasazení aplikací vytvořených pomocí Windows Forms nebo Windows Presentation Foundation (WPF). ClickOnce řeší tři problémy:

- Obtížné aktualizace nainstalované aplikace.
- Vliv aplikace na uživatelův počítač.
- Nutnost administrátorských oprávnění k nainstalování aplikace.

Aplikace nainstalované technologií ClickOnce jsou považovány za „low impact“ tj. aplikace s nízkým vlivem na operační systém a jiné nainstalované aplikace, protože jsou nainstalované pro konkrétního uživatele zvlášť a nejsou společné pro celý systém. Nejsou potřeba žádné administrátorské oprávnění pro instalování ClickOnce aplikací. Všechny aplikace ClickOnce jsou navzájem od sebe izolovány, takže nemůže dojít k narušení ostatních nainstalovaných aplikací.

ClickOnce aplikace mohou být nainstalovány na počítač z webu, sdíleného úložiště nebo např. CD. Aktualizace pomocí technologie ClickOnce je řízena XML manifest soubory (deployment manifest a application manifest). Deployment manifest popisuje možnost aktualizace, současnou verzi, identitu vydavatele a digitální podpis. Application manifest popisuje aplikační knihovny, závislé knihovny a list oprávnění vyžadovaný aplikací. K nainstalování aplikace musí uživatel kliknout na deployment manifest soubor. (Microsoft, 2013f)

ClickOnce umožňuje několik způsobů aktualizací aplikací:

- Kontrola nové verze aplikace před startem aplikace
- Kontrola nové verze aplikace po startu aplikace
- Vlastní kontrola naprogramována pomocí API ClickOnce

V ClickOnce je také zabudována podpora důležitých aktualizací, která zajišťuje, že všichni uživatelé aplikace budou aktualizovat na novou verzi v daném časovém horizontu.

Podpora ClickOnce je zabudována ve Visual Studio. Pokud programátor naprogramuje novou verzi aplikace, tak je mu umožněno novou verzi zpřístupnit uživatelům kliknutím na jediné tlačítko Publish. Aplikace se potom nahraje na sdílené úložiště nebo webový server dle zvoleného nastavení. Podle nastavených aktualizací má pak uživatel přístup k nové verzi, na kterou je upozorněn při startu aplikace.

S aktualizací aplikace souvisí i ukládání nastavení aplikace. Aktualizace si pamatuje poslední zvolenou šablonu pro certifikáty a dohody a poslední umístění kam byly vygenerovány, Setřídění seznamu školení a další. Microsoft umožňuje nastavení aplikace ukládat do XML souboru „app.config“. Jednotlivým položkám nastavení lze nastavit rozsah aplikace nebo rozsah uživatele. Rozdíl mezi uvedenými rozsahy spočívá v tom, že uživatelský rozsah je zachován při aktualizaci, kdežto aplikační rozsah je vždy po aktualizaci přepsán nastavením z nové verze. Mezi nastavení s aplikačním rozsahem patří connection string pro přístup k databázi MS ACCESS a connection string pro přístup MSSQL Serveru. Pokud by tedy došlo k přemístění databáze na jiný počítač nebo sdílené úložiště, stačí programátorovi vydat novou verzi s patřičně upraveným connection stringem a uživatel ani nepostřehne, že došlo k nějaké změně.

## Závěr

Cílem diplomové práce bylo navrhnout systém pro evidenci školení zaměstnanců a zároveň provést krátkou rešerši současných systémů evidence školení. Vyvinutá aplikace je nasazena ve firmě TRW, kde pomáhá personálnímu oddělení evidovat proškolené zaměstnance a zároveň poskytuje požadované výstupy pro půlroční monitorovací zprávy. Aplikaci využívají současně tři lidé, kteří naplňují databázi školení zaměstnanců údaji a využívají její výstupy.

Současné systémy evidence školení zaměstnanců jsou součástí komplexních personálních systémů, které poskytují rozsáhlé množství funkcí, modulů a konfigurací, aby mohly splnit co nejvíce požadavků firem. To je ale zároveň hlavním problémem těchto systémů. Uvedené systémy s rostoucí funkcionalitou přestávají být přehledné a stávají se složitější na pochopení. Nová funkcionalita nebo změna v systému vyžaduje drahocenný čas. Systémy neumožňují tvorbu certifikátů ani neposkytují všechny požadované výstupy pro monitorovací zprávy. Z uvedených důvodů vznikl požadavek na tvorbu aplikace evidence školení zaměstnanců, která bude jednoduchá na ovládání, přehledná a poskytne požadované výstupy.

Při tvorbě aplikace bylo nutné seznámit se celou řadou technologií a především s požadavky na výstupy a ovládání aplikace. K realizování diplomové práce bylo použito platformy .NET od společnosti Microsoft vzhledem k její robustnosti, popularitě, přístupné online dokumentaci, ale především proto, že ve firmě TRW je nasazen operační systém Microsoft Windows a platforma .NET je vyvíjena stejnou společností. Programování aplikace probíhalo v prostředí Microsoft Visual Studio pomocí jazyka VB.NET. Databáze byly navrženy v aplikacích MS ACCESS a SQL Server Management Studio. V prostředí relačních databází se nelze obejít bez znalosti jazyka SQL. Práce s databázemi vyžaduje na platformě .NET znalost technologie ADO.NET.

Aplikace evidence školení zaměstnanců dále umožňuje jednoduché generování a hromadný tisk certifikátů ze školení a dohod o účasti v projektu. V aplikaci lze snadno spočítat monitorovací indikátory pro potřebné monitorovací zprávy. Aplikace je automaticky aktualizována, pokud je k dispozici novější verze.

Aplikace evidence školení zaměstnanců dokáže k ukládání dat využívat dvě různé databáze MS ACCESS a MS SQL Server, což rozšiřuje možnosti nasazení a flexibilitu aplikace. Uložení dat do databáze umožňuje snadné rozšíření systému, případně může poskytnout přístup k datům i jiným aplikacím.

## Citace

FAIRSAIL. [online]. [cit. 2013-04-15]. Dostupné z: <http://www.fairsail.com/hr-cloud/hcm-social/>

GRIFFITHS, Ian a Matthew ADAMS. *.NET Windows forms in a nutshell*. 1st ed. Cambridge: O'Reilly, 2003, xi, 879 p. ISBN 05-960-0338-2.

KOLÁŘ, Petr. *Operační systémy* [online]. [cit. 2013-04-15]. Liberec: 2005-02-01. Dostupné z: <http://http://www.nti.tul.cz/~kolar/os/os.pdf>

LEISTNER. *Metodika monitorovacích indikátorů OP LZZ*. 2010-05-31. Dostupné z: [http://www.esfcr.cz/file/7840\\_1\\_1/](http://www.esfcr.cz/file/7840_1_1/)

MICROSOFT. *Visual Studio: Visual Studio Express 2012* [online]. [cit. 2013-04-15]. Dostupné z: <http://www.microsoft.com/visualstudio/eng/downloads#d-2012-express>

MICROSOFT. *Introduction to WPF* [online]. [cit. 2013-04-15]. Dostupné z: <http://msdn.microsoft.com/en-us/library/aa970268.aspx>

MICROSOFT. *Microsoft Microsoft® SQL Server® 2008 R2 SP2 - Express Edition* [online]. [cit. 2013-04-16]. Dostupné z: <http://www.microsoft.com/en-us/download/details.aspx?id=30438>

MICROSOFT. *ADO.NET Overview* [online]. [cit. 2013-04-16]. Dostupné z: <http://msdn.microsoft.com/cs-cz/library/h43ks021.aspx>

MICROSOFT. *ADO.NET Architecture* [online]. [cit. 2013-04-16]. Dostupné z: <http://msdn.microsoft.com/cs-cz/library/27y4ybxw.aspx>

MICROSOFT. [online]. [cit. 2013-04-18]. Dostupné z: <http://msdn.microsoft.com/en-us/library/142dbbz4%28v=vs.80%29.aspx>

NUGGET SW [online]. [cit. 2013-04-15]. Dostupné z: <http://www.nuggetsw.cz/koncepce-reseni/>

SHAFRANOVICH, Y. *Common Format and MIME Type for Comma-Separated Values (CSV) Files*. [online]. IETF, RFC 4180, Oct 2005. Dostupné z <http://tools.ietf.org/html/rfc4180>

SINGH, S. K. *Database Systems: Concept, Design and Applications*. 2nd ed. India: Pearson Education India, 2011. ISBN 8131760928.

THAI, Thuan L a Hoang Q LAM. *.NET framework essentials*. 3rd ed. Sebastopol, CA: O'Reilly, 2003, xiv, 363 p. ISBN 05-960-0505-9.

TOMADNLOVÁ Radka. ELM Základní informace. Interní zpráva firmy TRW. Jablonec nad Nisou, 2013

W3C. Extensible Markup Language (XML) [online]. 2012-01-24 [cit. 2013-04-16]. Dostupné z: <http://www.w3.org/XML/>



# Seznam příloh

## **OBSAH DVD**

- Zdrojové kódy aplikace evidence školení (VB.NET).
- Skripty pro tvorbu prázdné databáze na MSSQL 2008 R2 Express.
- Prázdná databáze MS ACCESS.
- Šablona certifikátu.
- Šablona dohody u účasti zaměstnance v projektu.
- Text diplomové práce ve formátu PDF.
- Návod k instalaci